Investigating the efficiency of fitting Coxian phase-type distributions to health care data

Payne, K., Marshall, A., & Cairns, K. (2012). Investigating the efficiency of fitting Coxian phase-type distributions to health care data. DOI: 10.1093/imaman/dpr008

Published in:
IMA Journal of Management Mathematics

Document Version:
Publisher's PDF, also known as Version of record

IMA Journal of Management Mathematics (2012) **23**, 133–145
doi:10.1093/imaman/dpr008
Advance Access publication on May 12, 2011

Investigating the efficiency of fitting Coxian phase-type distributions to health care data

KIERAN PAYNE*, ADELE H. MARSHALL AND KAREN J. CAIRNS
Centre for Statistical Science & Operational Research (CenSSOR), Queen's University, Belfast BT7 1NN, UK
*Corresponding author: k.payne@qub.ac.uk

[Received on 7 September 2010; accepted on 30 March 2011]

Coxian phase-type distributions are becoming a popular means of representing survival times within a health care environment. They are favoured as they show a distribution as a system of phases and can allow for an easy visual representation of the rate of flow of patients through a system. Difficulties arise, however, in determining the parameter estimates of the Coxian phase-type distribution. This paper examines ways of making the fitting of the Coxian phase-type distribution less cumbersome by outlining different software packages and algorithms available to perform the fit and assessing their capabilities through a number of performance measures. The performance measures rate each of the methods and help in identifying the more efficient. Conclusions drawn from these performance measures suggest SAS to be the most robust package. It has a high rate of convergence in each of the four example model fits considered, short computational times, detailed output, convergence criteria options, along with a succinct ability to switch between different algorithms.

Keywords: Coxian phase-type distributions; fitting; parameter estimation; optimization; health care data.

1. Introduction

Coxian phase-type distributions have been applied in many areas of research, including applications in queueing theory, supply chains, insurance and risk, telecommunication and health care (Cox, 1955; Feldmann & Whitt, 1998; Haddad *et al.*, 1998; Sasaki *et al.*, 2004; Vasilakis & Marshall, 2005). It has been previously shown that Coxian phase-type distributions are ideal in fitting survival data, particularly, patient length of stay within a health care environment. Faddy & McClean (1999) demonstrated within their analysis how Coxian phase-type distributions can be used to model the bed occupancy of male geriatric patients. The work of Marshall *et al.* (2002) describe how Coxian phase-type distributions can represent total time spent within a hospital accident and emergency department. The hospital length of stay for patients suffering from congestive heart failure has also been modelled using the Coxian phase-type distribution by Shaw & Marshall (2007).

Issues, however, arise when fitting the Coxian phase-type distribution, despite a reduction in the number of parameters to be estimated when compared to the general form of phase-type distributions (Neuts, 1981; Latouche & Ramaswami, 1999; Lang & Arthur, 1997; Faddy, 1998). The majority of parameter estimation algorithms require initial parameter estimates to be supplied. Within these algorithms, alterations are made to parameters in order to determine a set of resulting parameter estimates that maximize the log-likelihood function. The more parameters that have to be estimated, however, the more difficult this procedure becomes, increasing the complexity of the fitting process with the addition

of each new phase to the model. The process of choosing *good* initial estimates for the parameters, so that convergence be achieved also raises a number of difficulties (Marshall & Zenga, 2010).

The purpose of this investigation is to simulate data from a number of different Coxian phase-type distributions (which mimic the characteristics of a health care environment) and using a variety of different fitting techniques assess the ability of each technique to return to the original parameter values from which each simulation was derived.

Coxian data have been simulated, for a number of different sets of parameter values, with a number of different phases based on the cumulative distribution function (CDF) of the Coxian phase-type distribution. In order to assess the ability of each different software package, a variety of different performance measures will be considered to determine which results in a more accurate representation of the simulated data. Within each software package, a number of different algorithms may be considered (if available), with the performance measures used to illustrate any differences between the software packages examined and the methodologies considered. The MATLAB software package is commonly used for fitting the distributions using the Nelder–Mead (Nelder & Mead, 1965) algorithm by Fackrell (2009), Faddy & McClean (1999), Marshall *et al.* (2002), Marshall & Zenga (2010), Vasilakis & Marshall (2005). This work sets out to show that while MATLAB is both appropriate and accurate in what it does, it is not the only option available. By looking beyond the current methods at more recent developments in optimization, it is possible to obtain results which on occasion can be more accurate and better at representing the original population of a given dataset.

2. Coxian phase-type distribution

Neuts (1981) describes Coxian phase-type distributions as a special case of Markov model in which duration until an event occurs is expressed in terms of a process consisting of a sequence of latent phases. Based on the method of stages by A.K. Erlang and originating from the family of phase-type distributions, Coxian phase-type distributions are described as a versatile set of tractable models for applied probability (Neuts, 1981; Latouche & Ramaswami, 1999), which model random time intervals as being made up of a number of exponentially distributed segments (Fackrell, 2009).

The generality of phase-type distributions makes parameter estimation difficult. Coxian phase-type distributions are employed, as they describe duration until an event occurs as a process consisting of a sequence of latent phases. The process commences within the first phase and may either progress through the phases sequentially or enter into absorption, reducing the number of parameters required within the model to $2n - 1$, as illustrated in Fig. 1. The reduction of parameters within the model simplifies the process of calculating parameter estimates compared to phase-type distributions, however, for a large number of phases, the process of fitting the Coxian phase-type distribution still remains problematic (Faddy, 1998).

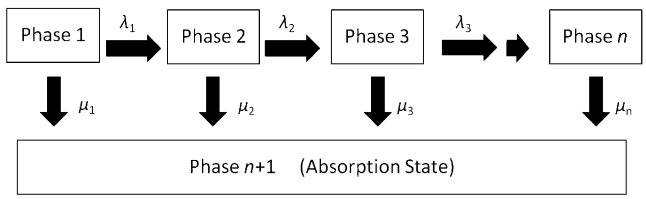

FIG. 1. General form of the *n*-phase Coxian phase-type model.

The Coxian phase-type distribution may be defined as follows: let $\{X(t); t \geqslant 0\}$ be a latent Markov chain in continuous time with states $\{1, 2, \ldots, n, n+1\}$, $X(0) = 1$ and for $i = 1, 2, \ldots, n$,

$$\text{prob}\{X(t+\delta t) = i+1 | X(t) = i\} = \lambda_i \delta t + \text{o}(\delta t), \tag{1}$$

$$\text{prob}\{X(t+\delta t) = n+1 | X(t) = i\} = \mu_i \delta t + \text{o}(\delta t), \tag{2}$$

where μ_i represents the rate of movement out of phase i to the absorbing state and λ_i the rate of movement from phase i to phase $i+1$, respectively (Cox & Miller, 1965; Neuts, 1981; Latouche & Ramaswami, 1999). The infinitesimal generator $\mathbf{Q}$ that describes movement through the model is defined as

$$\mathbf{Q} = \begin{pmatrix} -(\lambda_1+\mu_1) & \lambda_1 & 0 \ldots & 0 & 0 & \mu_1 \\ 0 & -(\lambda_2+\mu_2) & \lambda_2 \ldots & 0 & 0 & \mu_2 \\ \vdots & \vdots & \vdots \ddots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 \ldots & -(\lambda_{n-1}+\mu_{n-1}) & \lambda_{n-1} & \mu_{n-1} \\ 0 & 0 & 0 \ldots & 0 & -\mu_n & \mu_n \\ 0 & 0 & 0 \ldots & 0 & 0 & 0 \end{pmatrix}, \tag{3}$$

which can be summarized by defining the upper left $n \times n$ matrix of $\mathbf{Q}$ as $\mathbf{T}$:

$$\mathbf{Q} = \begin{bmatrix} \mathbf{T} & -\mathbf{Te}, \\ \mathbf{0} & 0 \end{bmatrix} \tag{4}$$

where $\mathbf{e}$ is an $n \times 1$ vector of ones and $\mathbf{0}$ a $1 \times n$ row vector of zeros. The probability density function $f(x)$ can then be represented by matrix notation in terms of $\mathbf{T}$.

$$\begin{aligned} f(x) &= \mathbf{p}\ \exp\{\mathbf{T}x\}\ \mathbf{q} \quad \text{for } x \geqslant 0, \\ \mathbf{p} &= (1\ 0\ 0 \ldots 0\ 0), \\ \mathbf{q} &= -\mathbf{Te} = (\mu_1\ \mu_2\ \ldots\ \mu_n)^\top. \end{aligned} \tag{5}$$

The CDF, $F(x)$, can be derived as

$$F(x) = 1 - (\mathbf{p}\ \exp\{\mathbf{T}x\}\ \mathbf{e}), \tag{6}$$

where the survival function, $S(x)$, is $(\mathbf{p}\ \exp\{\mathbf{T}x\}\ \mathbf{e})$.

3. Methodology

3.1 *Simulation of data*

A number of data sets from the Coxian phase-type distribution with known parameters need to be simulated. The parameter values chosen are carefully selected so that the shape of the underlying distribution within each data set mimics the common characteristics found within hospital data. That is, a highly skewed distribution with a peak located during earlier time points. Data have been simulated

TABLE 1 *Parameter values used to create simulated data*

Number of phases	Parameter values
1-Phase	$\mu_1 = 0.035$
3-Phase	$\mu_1 = 0.003$, $\mu_2 = 0.150$, $\mu_3 = 0.100$
	$\lambda_1 = 0.550$, $\lambda_2 = 0.050$
5-Phase	$\mu_1 = 0.020$, $\mu_2 = 0.100$, $\mu_3 = 0.120$, $\mu_4 = 0.050$, $\mu_5 = 0.750$
	$\lambda_1 = 0.750$, $\lambda_2 = 0.060$, $\lambda_3 = 0.500$, $\lambda_4 = 0.150$
7-Phase	$\mu_1 = 0.030$, $\mu_2 = 0.280$, $\mu_3 = 0.350$, $\mu_4 = 0.140$, $\mu_5 = 0.110$
	$\mu_6 = 0.650$, $\mu_7 = 0.140$ $\lambda_1 = 0.950$, $\lambda_2 = 0.750$, $\lambda_3 = 0.640$,
	$\lambda_4 = 0.810$, $\lambda_5 = 0.600$, $\lambda_6 = 0.200$

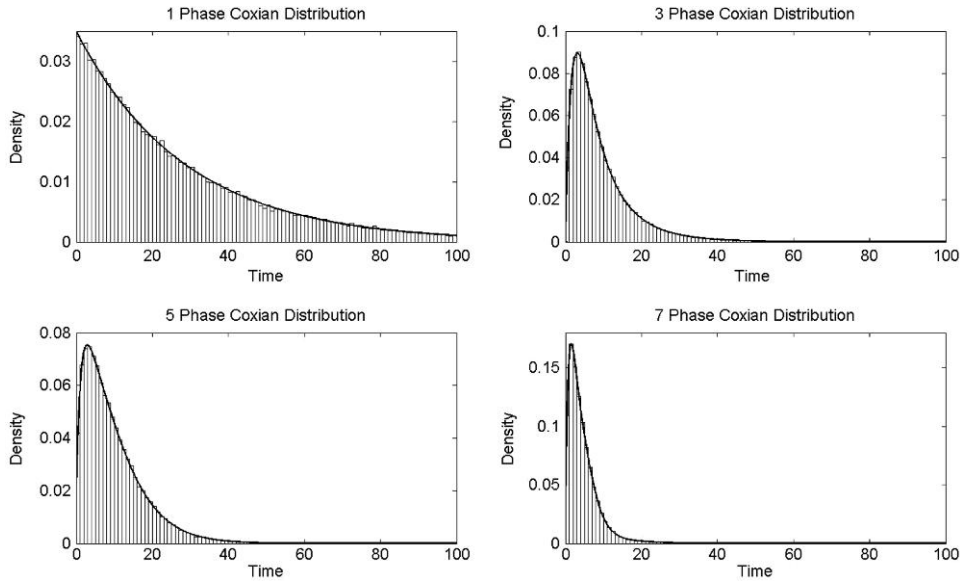

FIG. 2. The probability density function of the optimal parameter values obtained in optimization compared to the distribution of the simulated data.

for a 1-, 3-, 5- and 7-phase Coxian phase-type distribution to show how the complexity of fitting increases with the addition of new phases. A 1-phase distribution will contain one parameter value to be estimated, while a 7-phase distribution will require the successful estimation of 13 parameter values. The parameter values used are given in Table 1. The distribution of each of the data sets is shown in Fig. 2 and illustrates the skewed nature of health care data.

Each data set is created by examining the *CDF* of the Coxian phase-type distribution shown in equation 6. Random numbers are first generated from the uniform [0,1] distribution, (100,000 for this analysis), and the cdf is then used to determine the corresponding time associated with each random number. A dataset of 100,000 time values is therefore simulated for each of the 1-, 3-, 5- and 7-phase Coxian distributions as listed in Table 1.

3.2 *Numeric algorithms & software packages*

The aim of this analysis is to explore the options available when fitting Coxian phase-type distributions. By taking each of the data sets outlined in Section 3.1, Coxian phase-type distributions will be fitted using a variety of different software options and algorithms. Five different software environments have been considered:

- MATLAB—Optimization toolbox (MATLAB, 2010).
- NAG (Numerical Algorithm Group)—Toolbox for MATLAB.
- SAS–Proc NLP (Non-Linear Programming) procedures (SAS, 2008).
- R–Optim (Optimization) package & NLM (Non-Linear Minimization) package (R Foundation for Statistical Computing, 2009).
- EMpht—Expectation–Maximization (EMpht, 1998).

Within these packages, a number of different algorithms are available. Not all algorithms are available for all software environments, however, across all software environments the following algorithms are considered:

1. Nelder–Mead simplex algorithm—proposed by John Nelder & Roger Mead in 1965 the Nelder–Mead simplex method is an extension of the simplex method of Spendely, Hext and Himsworth (Nelder & Mead, 1965; Bunday, 1984).
2. Quasi-Newton algorithm—based on Newtons method, this method has the advantage that the Hessian matrix is updated on each iteration by analysing successive gradient vectors instead of performing a recalculation.
3. Trust region algorithm—iteratively optimizes a quadratic approximation to the non-linear objective function within a hyper-elliptic trust region (Dennis *et al.*, 1981; Gay, 1983; Moré & Sorensen, 1983).
4. Newton–Raphson algorithm with line search—this algorithm uses a pure Newton step when the Hessian is positive definite and the Newton step reduces the value of the objective function successfully. Otherwise, a combination of ridging and linesearch is done to compute successful steps. If the Hessian is not positive definite, a multiple of the identity matrix is added to the Hessian to make it positive. Within each iteration, a linesearch is done along the search direction to find an approximate optimum of the objective function.
5. Newton–Raphson algorithm with ridging—uses a pure Newton step when the Hessian is positive definite and when the Newton step reduced the value of the objective function successfully. If at least one of these two conditions is not satisfied, a multiple of the identity matrix is added to the Hessian matrix. As this method uses orthogonal decomposition of the approximate Hessian, the time to perform each iteration may be slower than other algorithms.
6. Double dogleg algorithm—combining the idea of the quasi-Newton algorithm along with the trust region algorithm, the double dogleg algorithm computes in each iteration the step $s^{(k)}$ as the linear combination of the steepest descent or ascent search direction $s_1^{(k)}$ and a quasi-Newton search direction $s_2^{(k)}$ such that $s^{(k)} = \alpha_1 s_1^{(k)} + \alpha_2 s_2^{(k)}$.
7. Conjugate gradient method—during n successive iterations, uninterrupted by restarts or changes in the working set, the conjugate gradient algorithm computes a cycle of n conjugate search directions in each iteration, a linesearch is done along the search direction to find an approximate

optimum of the objective function. The default search method uses quadratic interpolation and cubic extrapolation to obtain a step size α satisfying *Goldstein's* condition.

8. Expectation–maximization algorithm—an iterative method for maximum likelihood estimation, it alternates between performing an expectation (E) step, which computes the expectation of the log-likelihood evaluated using the current estimate for the latent variables, and a maximization (M) step, which computes parameters maximizing the expected log-likelihood found in the E step. These parameter estimates are then used to determine the distribution of the latent variables in the next E step (Asmussen *et al.*, 1996).

3.3 *Initial parameter estimates*

All the numerical optimization algorithms considered require the user to provide some feasible starting values for the parameters to be estimated. In general, the closer the initial parameter values are to the true parameters of the data, the quicker the algorithm will reach convergence. From visual representation, it seems apparent that the parameter values would not exceed one and by definition cannot be less than zero.

The mathematical software package, Wolfram Mathematica (0000) is used to provide a visual aid from which initial parameter estimates can be developed. Using Mathematica, a plot of the underlying distribution of each data set is obtained, and then superimposed upon this distribution is the probability density function of the Coxian phase-type distribution. The parameter values, which have the effect of changing the shape of the probability density function, are then varied using a dynamic sliding scale within the Mathematica environment until, by visual inspection, the parameter values look to be a suitable approximation to the underlying distribution.

Fifty sets of initial parameter estimates are then generated based on random observations from the beta distribution, with mean corresponding to the values suggested by visual inspection from Mathematica. These resulting parameter estimates are then used as initial inputs to the optimization routines to fit the Coxian phase-type distribution to the data.

3.4 *Performance measures*

To assess the accuracy of each method, four performance measures are considered, three of these are based on assessing the accuracy of the algorithms, while the fourth addresses the relative speed.

1. Rate of convergence (ROC)—a measure of the number of successful convergences out of the total number of attempted fits *(s)*.

$$\mathrm{ROC} = \frac{\sum_{i=1}^{s} \delta_i}{s}, \tag{7}$$

where $\delta_i = 1$ indicates a successful convergence and $\delta_i = 0$ indicates unsuccessful convergence.

2. Mean relative distance (MRD)—a measure of how close the resulting parameter values are to the true values used in simulating the data for each attempted fit i:

$$\mathrm{MRD}_i = \frac{\sum_{p=1}^{n} \frac{|\mu_p - \mu_{pi}|}{\mu_p} + \sum_{p=1}^{n-1} \frac{|\lambda_p - \lambda_{pi}|}{\lambda_p}}{(2n-1)} \quad \text{for } i = 1, \ldots, s, \tag{8}$$

where the known parameter values of the Coxian phase-type distribution are denoted ($\mu_1, \mu_2, \ldots, \mu_n, \lambda_1, \ldots, \lambda_{(n-1)}$) and the parameter estimates obtained for the ith fit are denoted ($\mu_{1i}, \mu_{2i}, \ldots, \mu_{ni}, \ldots, \lambda_{1i}, \lambda_{(n-1)i}$).

- Average mean relative distance

$$\text{Average MRD} = \frac{\sum_{i=1}^{s} \delta_i \text{MRD}_i}{\sum_{i=1}^{s} \delta_i}. \tag{9}$$

- Minimum mean relative distance

$$\text{Minimum MRD} = \min_i (\text{MRD}_i). \tag{10}$$

3. Difference in area—a calculation of the area between the known probability density function and the probability density function from the resulting parameter values obtained in fit i.

$$\text{Difference in area} = \int_0^\infty |f(x|\mu_{1i}, \ldots, \lambda_{(n-1)i}) - f(x|\mu_1, \ldots, \lambda_{(n-1)})| \text{dx}. \tag{11}$$

4. Timing—a measure of the average (real) time an optimization algorithm takes to reach successful convergence.

4. Results

4.1 *Measures of accuracy*

4.1.1 *Fitting the 1-Phase Coxian distribution.* Table 2 displays the results for fitting the 1-phase distribution. A high rate of convergence is noted. There is consistency in the results of the average MRD, minimum MRD, average area and minimum area across all methods, indicating accuracy, with the exception of EMpht. On all occasions, EMpht was capable of reaching the known parameter values exactly.

Note that the package OPTIM in R cannot be used for fitting the 1-phase distribution. This is because the package is a general purpose optimization package for problems involving more than one parameter. As the Coxian phase-type distribution has only one unknown parameter when considering a 1-phase model, the package is unsuitable.

The exercise of fitting an exponential distribution was included to illustrate that although specific parameter values were chosen from which to simulate a data set, the simulation process may produce a resulting data set whose 'true' parameter value may be very slightly different

4.1.2 *Fitting the 3-Phase Coxian distribution.* Extending the Coxian phase-type distribution to 3-phases introduces five parameter values. Making reference to Table 3, it is again noted that across the majority of methods, there is a high rate of convergence. The algorithms implemented through the software package R do not perform as well with OPTIM having a 56% rate of convergence and NLM 48%, but of those which do converge performance is acceptable; particularly, when compared to the

TABLE 2 *Performance measures of different software/algorithm combinations for fitting 1-phase Coxian distribution*

Method	ROC (%)	Average MRD	Minimum MRD	Average Area	Minimum Area
MATLAB - fminsearch	100	0.000872	0.000839	0.000646	0.000621
NAG - E04JY	100	0.000872	0.000872	0.000646	0.000646
SAS - TRUREG	92	0.000872	0.000872	0.000645	0.000645
SAS - NEWRAP	100	0.000872	0.000872	0.000645	0.000645
SAS - NRRIDG	98	0.000872	0.000871	0.000645	0.000644
SAS - QUANEW	100	0.000872	0.000872	0.000644	0.000644
SAS - DBLDOG	100	0.000872	0.000871	0.000645	0.000643
SAS - CONGRA	100	0.000846	0.000676	0.000625	0.000499
SAS - NMSIMP	100	0.000873	0.000864	0.000645	0.000639
R - NLM	98	0.000876	0.000870	0.000649	0.000644
EMpht	100	0	0	0	0

TABLE 3 *Performance measures of different software/algorithm combinations for fitting 3-phase Coxian distribution*

Method	ROC (%)	Average MRD	Min MRD	Average Area	Minimum area
MATLAB - fminsearch	88	1.57×10^{15}	1.09×10^{-2}	0.024121077	0.001066083
NAG - E04JY	94	1.57×10^{3}	1.09×10^{-2}	0.019431800	0.001065372
SAS - TRUREG	100	6.28×10^{-1}	1.08×10^{-2}	0.018362572	0.001062758
SAS - NEWRAP	100	7.44×10^{-1}	1.08×10^{-2}	0.015623860	0.001053666
SAS - NRRIDG	100	6.66×10^{-1}	1.07×10^{-2}	0.016939695	0.001061336
SAS - QUANEW	100	8.74×10^{-1}	9.94×10^{-3}	0.016718916	0.001046326
SAS - DBLDOG	100	6.62×10^{-1}	4.86×10^{-3}	0.015757287	0.000899504
SAS - CONGRA	100	1.63×10	6.30×10^{-3}	0.030984880	0.000860732
SAS - NMSIMP	100	3.02×10	1.09×10^{-2}	0.029922630	0.001064852
R - OPTIM	56	1.49×10	1.19×10^{-2}	0.008427766	0.000973142
R - NLM	48	1.30×10	1.05×10^{-2}	0.005756146	0.001049133
EMpht	100	7.24×10^{-1}	1.16×10^{-2}	0.006553818	0.001044674

results of other methods. The smallest average area is given by the R NLM method. MATLAB produced some parameter estimates that varied greatly in comparison to the known parameter values. Given how MRD is calculated, a percentage difference between two values, when the resulting parameter value varies greatly in comparison to the known parameter value, the resulting MRD can be quite large. It is for this reason, a large average MRD is observed within the MATLAB results. Some of the resulting values for λ_2 varied significantly to the true value, while still allowing convergence to be achieved. The minimum MRD however is in line with the results from other algorithms, indicating that only some of the initial parameter values caused the algorithm difficulty. The method that produced the smallest minimum area, for this particular 3-phase Coxian distribution, is the SAS conjugate gradient algorithm. Figure 2 shows how the optimal parameter values from optimization compare when fitted against the underlying data set. It can be seen for the 3-phase distribution that the resulting parameter values provide an excellent representation of the simulated data.

4.1.3 *Fitting the 5-Phase Coxian distribution.* As more phases are introduced to the model, the more complex the problem becomes, with some methods beginning to perform better than others. Table 4 shows the results of fitting the 5-phase Coxian distribution that requires nine parameter values to be estimated. The R package NLM was unable to provide any meaningful results, given the size of the data and complexity of the problem and has therefore been left out of the analysis at this stage. EMpht did run, however, the time to reach convergence was considerably longer than that of other methods, a point that shall be discussed in more detail later. It is noted again that the average MRD is quite consistent with only the NAG quasi-Newton algorithm failing, on average, to produce parameter values close to the true parameters. The average area is small throughout all methods, with the smallest minimum area coming from the SAS conjugate gradient algorithm and double dogleg algorithm.

4.1.4 *Fitting the 7-Phase Coxian distribution.* On consideration of the largest model, Table 5, it is possible to examine each method, to determine which are consistently good at obtaining parameter

TABLE 4 *Performance measures of different software/algorithm combinations for fitting 5-phase Coxian distribution*

Method	ROC (%)	Average MRD	Minimum MRD	Average area	Minimum area
MATLAB - fminsearch	98	2.22×10	3.62×10^{-1}	0.004328519	0.002724926
NAG - E04JY	96	1.98×10^{37}	7.77×10^{-1}	0.018186271	0.003048986
SAS - TRUREG	100	1.40×10	2.63×10^{-1}	0.006563658	0.002819470
SAS - NEWRAP	90	1.31×10	2.59×10^{-1}	0.006037221	0.002758117
SAS - NRRIDG	100	1.29×10	2.48×10^{-1}	0.007428773	0.003104434
SAS - QUANEW	100	1.39×10	2.76×10^{-1}	0.008235490	0.002991250
SAS - DBLDOG	100	1.20×10	3.17×10^{-1}	0.007823264	0.001852684
SAS - CONGRA	86	1.19×10	3.43×10^{-1}	0.005810419	0.001836278
SAS - NMSIMP	86	1.87×10	3.82×10^{-1}	0.007428773	0.003104434
R - OPTIM	96	2.06×10	6.60×10^{-1}	0.005361697	0.002786863
EMpht	100	1.39×10	3.72×10^{-1}	0.003312631	0.002991728

TABLE 5 *Performance measures of different software/algorithm combinations for fitting 7-phase Coxian distribution*

Method	ROC (%)	Average MRD	Minimum MRD	Average area	Minimum area
MATLAB - fminsearch	84	1.24×10^{181}	1.85×10^{-1}	0.015603043	0.004456644
NAG - E04JY	100	5.12×10^{30}	1.98×10^{-1}	0.029882019	0.004534112
SAS - TRUREG	96	5.21×10^{-1}	1.83×10^{-1}	0.034299521	0.004074434
SAS - NEWRAP	68	4.08×10^{-1}	9.15×10^{-2}	0.013407281	0.004077839
SAS - NRRIDG	92	4.38×10^{-1}	1.28×10^{-1}	0.017280863	0.004217608
SAS - QUANEW	100	4.65×10^{-1}	1.12×10^{-1}	0.009883611	0.003947655
SAS - DBLDOG	96	5.21×10^{-1}	1.83×10^{-1}	0.034299521	0.004074434
SAS - CONGRA	98	4.48×10^{-1}	1.22×10^{-1}	0.008917830	0.004050693
SAS - NMSIMP	20	5.22×10^{-1}	1.87×10^{-1}	0.009502400	0.004435215
R - OPTIM	90	9.00×10^{-1}	1.91×10^{-1}	0.013571704	0.004198692

values true to the given data. The 7-phase model requires 13 parameter values to be selected in such a way as to maximize the log-likelihood function so the resulting parameters are a true and accurate representation. For the same reasons as mentioned previously, R-NLM is not considered. As EMpht was taking a considerably long length of time to reach convergence for a single set of parameter estimates (>24 h), it was felt that it would not be an appropriate use of machine time to complete all 50 sets of initial parameter estimates.

As previous the NAG quasi-Newton algorithm has a large average MRD, indicating that a number of parameter results are not close to the true values. This is also noted in MATLAB's fminsearch, with an average MRD of $1.241 \times ^{181}$. All the algorithms implemented through SAS have a low average MRD along with the R-NLM package. For the 7-phase distribution, the SAS Nelder–Mead simplex method has a low rate of convergence with only 20% reaching convergence.

4.2 *Timings*

As discussed in Section 3, the time for each method to reach convergence is considered an important measure. Considerable differences were noted between methods when carrying out the fitting process. To further examine these differences, timings were recorded for each of the methods when fitting the 3-phase Coxian phase-type distribution model. The results, which illustrate the average time it took each method to reach successful convergence from a single set of initial parameter estimates, are shown in Fig. 3.

We make note firstly to the scale of the *y* axis. The time for each method has been plotted on a logarithmic scale due to such considerable differences. MATLAB and NAG took on average 529 and 357 s to reach a successful convergence. This is not as quick as the R methods that took on average 236 s for the Optim package and 269 s for the NLM package. However, when compared with EMpht, which took on average 2194 s (circa 37 min), there is a considerable difference in the time to reach successful convergence. None of these methods, however, were able to outperform SAS. The average time it took to reach successful convergence for a SAS method varied from 8 s (SAS–double dogleg algorithm) to

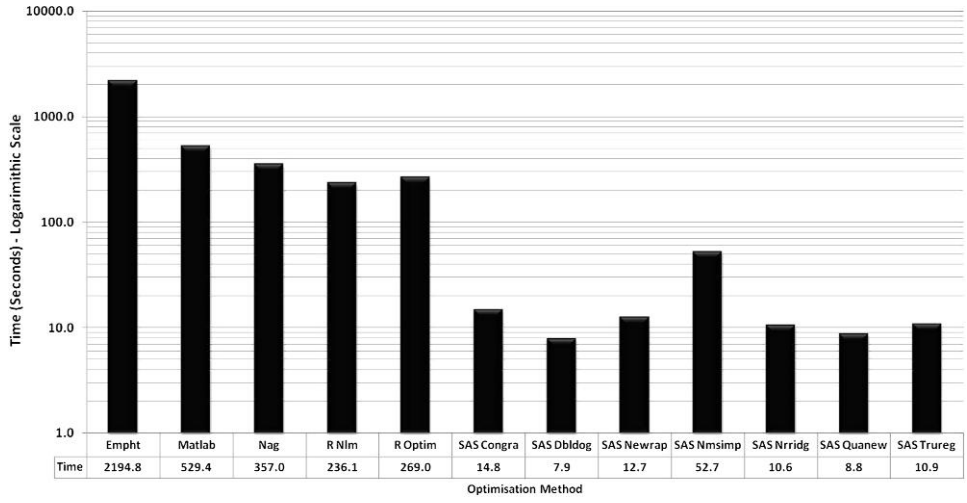

FIG. 3. Average time to reach convergence for multiple methods when fitting a 3-phase Coxian distribution.

53 s (SAS Nelder–Mead simplex algorithm). A 273% decrease in time from the slowest method EMpht, compared to the fastest SAS–double dogleg. Information on timings has only been presented for the 3-phase distribution, however, the order of timings remained consistent in the higher phases with SAS always achieving convergence first, and EMpht taking the longest.

5. Discussion

Reflecting on each of the methods examined, and the objectives set out in the introduction, it is evident that all the algorithms are capable of successfully fitting the Coxian phase-type distribution up to a 3-phase distribution. While it is believed that successful convergence could be achieved for some of the methods that have not been considered in the higher phases, this could only be achieved by allowing the algorithms a prolonged period of computational time—a cost that outweighs the benefit given the availability of other more successful options.

While R Optim was not applicable in fitting a 1-phase distribution, it was successful in all other phases and outperformed the R NLM package in terms of the number of successful convergences. The R NLM package struggled to converge with only a 48% convergence rate in the 3-phase model and when more parameters were introduced in the higher phases it was not possible to obtain successful convergences.

MATLAB and NAG did consistently have a higher rate of convergence but also produced on occasion a very large average MRD, indicating that convergence was reached but the resulting parameter estimates differed greatly from the known parameter values.

EMpht was considered to be very successful but computationally slower than the other methods, as illustrated in Fig. 3. With the 5-phase distribution taking over 24 h to reach a converged set of parameter values, it was not feasible to fit the 7-phase distribution.

SAS therefore became the software package of choice. It had a high rate of convergence and produced a consistently low average MRD across all the algorithms implemented through the software. The time to reach convergence was also considerable faster than the other methods. SAS is also favoured because of its flexibility and detailed output. Within one software environment, seven different algorithms can be implemented by making minimal changes to code, detailed output is provided for each optimization including the approximate standard error of the resulting parameter estimates including associated p value. The Hessian matrix and projected Hessian matrix are provided along with the covariance matrix and approximate correlation matrix providing a much more detailed and comprehensive output. SAS is not, however, a free software package, in comparison to R and is therefore not open source.

These conclusions have been based upon the results shown within this paper in which only four model fits have been presented. Potentially, the results may differ if different examples had been considered. However, while no formal testing has been done on other examples, user experience suggests that the results of further testing would not contradict these findings.

Further work could include examining the importance of initial parameter values used in each of the algorithms and software packages. Here, only structured initial values have been considered, however, some of the algorithms or software packages considered may be more suitable when initial parameters values are not sensibly chosen.

Funding

Northern Ireland Department for Employment and Learning (DEL).

REFERENCES

ASMUSSEN, S., NERMAN, O. & OLSSON, M. (1996) Fitting phase-type distributions via the EM algorithm. *Scand. J. Stat.*, **23**, 419–441.

BUNDAY, B. D. (1984) *Optimisation Methods*. London: Edward Arnold.

COX, D. R. (1955) On the use of complex probabilities in the theory of stochastic processes. *Proc. Camb. Philos. Soc.*, **51**, 313–319.

COX, D. R. & MILLER, H. D. (1965) *The Theory of Stochastic Processes*. London: Methuen.

DENNIS, J. E., GAY, D. M. & WELSH, R. E. (1981) An adaptive nonlinear least-squares algorithm. *AMC Trans. Math. Softw.*, **7**, 348–368.

EMPHT (1998) *Expectation Maximisation Package*. Available at http://home.imf.au.dk/asmus/pspapers.html. Accessed July 2010.

FACKRELL, M. (2009) Modelling health care systems with phase-type distributions. *Health Care Manag. Sci. Springer*, **12**, 11–26.

FADDY, M. (1998) On inferring the number of phases in a Coxian phase-type distribution. *Commun. Stat. Stoch. Model.*, **14**, 407–417.

FADDY, M. J. & MCCLEAN, S. I. (1999) Analysing data on lengths of stay of hospital patients using phase-type distributions. *Appl. Stoch. Model. Bus. Ind.*, **15**, 311–317.

FELDMANN, A. & WHITT, W. (1998) Fitting mixtures of exponentials to long-tail distributions to analyse network performance models. *Perform. Eval.*, **32**, 245–279.

GAY, D. M. (1983) Subroutines for unconstrained minimisation. *ACM Trans. Math. Softw.*, **9**, 503–524.

HADDAD, S., MORWAUX, P. & CHIOLA, G. (1998) Cox and phase-type distributions in stochastic Petri-nets—efficient derivation of solutions. *RAIRO Recherche Operationnelle-Oper. Res.*, **32**, 289–323.

LANG, A. & ARTHUR, J. L. (1997) Parameter approximation for phase-type distributions. *Matrix-Analytic Methods in Stochastic Models* (S. R. Chakravarthy & A. S. Alfa eds), vol. 189. New York: Marcel Dekker, pp. 151–206.

LATOUCHE, G. & RAMASWAMI, V. (1999) *Introduction to Matrix Analytic Methods in Stochastic Modelling*. Philadelphia, PA: ASA & SIAM.

MARSHALL, A. & ZENGA, M. A. (2010) Experimenting with the Coxian phase-type distribution to uncover suitable fits. *Methodol. Comput. Appl. Probab.*, **14**, 71–86.

MARSHALL, A. H., BURNS, L. & SHAW, B. (2002) Patient activity in hospital using discrete conditional phase-type (DC-PH) models. *Recent Advances in Stochastic Modelling & Data Analysis* (C. H. Skiados ed.). World Scientific, 2007, pp. 154–161.

MATLAB (2010) *The MathWorks, Optimisation Toolbox 5 User Guide*. Available at http://www.mathworks.com/help/toolbox/optim. Accessed July 2010.

MORÉ, J. J. & SORENSEN, D. C. (1983) Computing a trust-region step. *SIAM J. Sci. Stat. Comput.*, **4**, 553–572.

NAG—NUMERICAL ALGORITHM GROUP. *NAG Toolbox for MATLAB Mark 22 User Guide*. Available at http://www.nag.co.uk/numeric/MB/start.asp.

NELDER, J. A. & MEAD, R. (1965) A simplex method for function minimisation. *Comput. J.*, **7**, 308.

NEUTS, M. F. (1981) *Matrix-Geometric Solutions in Stochastic Models—An Algorithmic Approach*. Baltimore, MD: John Hopkins University Press.

R FOUNDATION FOR STATISTICAL COMPUTING (2009) *R: A Language and Environment for Statistical Computing*. Vienna, Austria: R Foundation for Statistical Computing. Available at http://www.R-project.org. Accessed July 2010.

SAS/OR(R) (2008) *9.2 User Guide: Mathematical Programming, The NLP Procedure*. Mathematica, Version 7.0. Champaign, IL: Wolfram Research, Inc. Available at http://www.sas.com. Accessed July 2010.

SASAKI, Y., IMAI, H., TSUNOYAMA, M. & ISHII, I. (2004) Approximation of probability distribution functions by Coxian distribution to evaluate multimedia systems. *Syst. Comput. Jpn.*, **35**, 16–24.

SHAW, B. & MARSHALL, A. H. (2007) Modelling the flow of congestive heart failure patients through a hospital system. *J. Oper. Res. Soc.*, **58**, 212–218.

VASILAKIS, C. & MARSHALL, A. H. (2005) Modelling nationwide hospital length of stay: opening the black box. *J. Oper. Res. Soc.*, **56**, 862–869.

WOLFRAM MATHEMATICA (7) (0000) Available at (http://www.wolfram.com/products/mathematica/index.html. Accessed July 2010).